

In the United States Patent and Trademark Office

In re the Application of:

Yen-Fu Chen)

Serial Number: 10/692,173)

Group: 2173

Docket Number: AUS920030664US1)

Examiner: Haoshin Shih

Filed on: 10/23/2003)

For: "System and Method for Automatic)

Information Compatibility Detection and)

Pasting Intervention")

REPLACEMENT APPEAL BRIEF

Real Party in Interest per 37 CFR §41.37(c)(1)(i)

The subject patent application is owned by International Business Machines Corporation of Armonk, NY.

Related Appeals and Interferences per 37 CFR §41.37(c)(1)(ii)

The present patent application is related to US Patent Application number 10/455,159, which has issued as US 7,310,781.

Status of Claims per 37 CFR §41.37(c)(1)(iii)

Claims 1 - 15 are finally rejected. The rejections of Claims 1 - 15 are appealed.

Status of Amendments after Final Rejections per 37 CFR §41.37(c)(1)(iv)

No amendments to the claims have been submitted or entered after final rejections.

Summary of the Claimed Subject Matter per 37 CFR §41.37(c)(1)(v)

Our invention provides an automatic compatibility check when a user tries to "paste" an item from a clipboard buffer to another computer resource, such as a file which the user is editing. In a first unique aspect, the invention follows a destination-first, source-second user interface model of copying and pasting, which is opposite of the normal approach. For example, instead of first copying the source information from the source file, our invention allows the user to initially designate the destination point in the destination file. This is useful because the user is usually already in the GUI for the destination file when he or she realizes some information needs to be inserted from somewhere else. Then, instead of having the user select the destination second, our invention allows the user to select the source information second. So, the process is opposite of the normal cut-and-paste approach – instead, it is almost a paste-and-cut approach. We find that this flow is much more natural and efficient in most circumstances.

A second unique aspect of our invention is that it automatically provides a compatibility check between the copied information and the destination file. For example, if the destination file is a spreadsheet which is incapable of handling video clip objects, and the user tries to paste a video clip object into the destination file, a compatibility check will catch the incompatibility, and a handler will be launched. The handler can simply notify the user of the incompatibility, or perform a conversion of some sort.

Claim 1 is directed towards a method according to the invention "for transferring content from one computer resource to another computer resource" by:

performing by a user in a first user interface to a first computer resource a designation of a destination point or area in said first computer resource (*paras. 0099, 0150; Fig. 5 #51; Fig. 7 #104; etc.*);

subsequently, performing by said user in a second user interface to a second computer resource a selection of two or more information elements in said second computer resource (*para. 0101; Fig. 5 #56; Fig. 7 #103; etc.*);

subsequent to said step of selection, automatically copying (*paras. 0087, 0102*) said selected information elements into a transfer buffer, thereby concatenating (*paras. 0056, 0088, 0100, 0104, etc.*) two or more information elements into said buffer, said transfer buffer comprising a clipboard in memory (*paras. 0087, 0111, 0117; Fig. 7*

clipboard icon);

upon attempt to automatically transfer said information items from said buffer, intercepting the transfer to said destination point or area of one or more information elements (*paras. 0103, 0144; Fig. 5 #501*);

performing a compatibility check for each intercepted information element with the destination computer resource by consulting one or more user-configurable compatibility rules to classify elements as incompatible or compatible (*paras. 0144 - 0151; all of Fig. 6*);

for each incompatible element, performing a compatibility handling action as defined by one or more conversion rules (*para. 0147; Table 2 <action> . . . </action>*); and

for each compatible element, allowing transfer of the unmodified compatible element to the destination (*para. 0146*).

Claim 6 is directed towards a computer readable medium (*examples set forth in paras. 0062 - 0064*) with software for causing a processor to perform a method according to the invention:

performing by a user in a first user interface to a first computer resource a designation of a destination point or area in said first computer resource (*paras. 0099, 0150; Fig. 5 #51; Fig. 7 #104; etc.*);

subsequently, performing by said user in a second user interface to a second computer resource a selection of two or more information elements in said second computer resource (*para. 0101; Fig. 5 #56; Fig. 7 #103; etc.*);

subsequent to said step of selection, automatically copying (*paras. 0087, 0102*) said selected information elements into a transfer buffer, thereby concatenating (*paras. 0056, 0088, 0100, 0104, etc.*) two or more information elements into said buffer, said transfer buffer comprising a clipboard in memory (*paras. 0087, 0111, 0117; Fig. 7 clipboard icon*);

upon attempt to automatically transfer said information items from said buffer, intercepting the transfer to said destination point or area of one or more information elements (*paras. 0103, 0144; Fig. 5 #501*);

performing a compatibility check for each intercepted information element with the destination computer resource by consulting one or more user-configurable compatibility rules to classify elements as incompatible or compatible (*paras. 0144 - 0151; all of Fig. 6*);

for each incompatible element, performing a compatibility handling action as defined by one or more conversion rules (*para. 0147; Table 2 <action> . . . </action>*); and

for each compatible element, allowing transfer of the unmodified compatible element to the destination (*para. 0146*).

Claim 11 is directed towards a system according to the invention, having:

a transfer buffer comprising a clipboard in memory (*paras. 0087, 0111, 0117; Fig. 7 clipboard icon*);

a destination-first, source-second element copier configured to allow a user in a first user interface to a first computer resource to designate a destination point or area in said first computer resource, to subsequently select in a second user interface to a second computer resource two or more information elements in said second computer resource, and to subsequently automatically copy said selected information elements into said transfer buffer thereby concatenating said information elements into said buffer (*paras. 0056, 0087 - 0088, 0099, 0100 - 0104, 0117, 0150; Fig. 5 #51, #56; Fig. 7 #103, #104, clipboard icon*);

a transfer interceptor configured to, upon attempt to copy said information elements from said buffer, intercept one or more information elements (*paras. 0103, 0144; Fig. 5 #501*);

a compatibility checker configured to verify the compatibility of each intercepted information element with the destination by consulting one or more user-configurable compatibility rules to classify elements as incompatible or compatible (*paras. 0144 - 0151; all of Fig. 6*); and

a compatibility action handler configured to perform an action as defined by one or more conversion rules for each intercepted information element, and further configured to allowing transfer of said compatible elements to the destination without

modification (*paras. 0146 - 0147; Table 2* *<action> . . . </action>*).

Grounds for Rejection For Which Review is Sought per 37 CFR §41.37(c)(1)(vi)

Review by the Board of the rejections of 1, 5 - 6, 10 - 11 and 15 under 35 U.S.C. §103(a) over Stern *et al.* (Stern, US 6,807,668 B2) and Apperley *et al.* (Apperley, "Breaking the copy/paste cycle: the stretchable selection tool"), hereinafter referred to as "Stern in view of Apperley".

Review is also sought of the rejections of Claims 2 - 4, 7 - 9, and 12 - 14 under 35 U.S.C. §103(a) over Stern in view of Apperley in further view of Tomm *et al.* (Tomm, US 6,560,608 B1) and Tsuji *et al.* (Tsuji, US 5,586,025), hereinafter referred to as Stern in view of Apperley, Tomm and Tsuji.

Arguments per 37 CFR §41.37(c)(1)(vii)

We respectfully believe that errors in examination have occurred in the rejections of our claims in the following manners:

- (a) We believe a *prima facie* case of obviousness has not been properly established for failure to establish what was the ordinary skill level in the art at the time of our invention. Although the Examiner has made no explicit comments as to the ordinary skill level used for the obviousness analysis, we believe that the cited references cannot be relied upon as an indication of ordinary skill level because the authors and inventors of the references are extraordinarily skilled as evidenced by likely over 100 patents and scholarly works of the authors and inventors.
- (b) We believe a *prima facie* case of obviousness has not been properly established because there can be no obviousness where one or more references teach away from the proposed combination or modifications. We believe the Apperley references teaches away from the proposed combination with the Stern reference.
- (c) We believe that Stern in view of Apperley in further view of Tomm and in still further view of Tsuji fail to teach all of our claimed steps, elements and limitations, especially "concatenation" of information in a clipboard.
- (d) We believe improper use of hindsight to examine our claims in a piecemeal fashion through considering only the differences between the cited art and the claims, without considering our claims as a whole, may have occurred, as evidenced by the large number of references applied to the rejections of the claims (e.g. four references for about as many elements or steps).

Rejections under 35 U.S.C. §103(a) Stern in view of Apperley

Error for Failing to Established Ordinary Skill Level. In our first reply to these rejections, we requested the Examiner to explicitly establish what skill level was used as the "ordinary skill level in the art at the time of our invention" for the obviousness analysis. In the final rejections, the Examiner has reposed our request by describing *what steps* would have been obvious to do, but without stating *who* would have been able and motivated to do this in terms of *skill level*. We believe this response this does not establish a *skill level* using any terms related to artisans's technical characteristics, such as those criteria (e.g. degrees held, number years of experience, etc.) suggested by the Court to determine the ordinary skill level (*Environmental Designs, Ltd. v. Union Oil*, 713 F.2d 693, 696, 218 USPQ 865, 868 (Fed. Cir. 1983); *Bausch & Lomb, Inc. v. Barnes-Hind/Hydrocurve, Inc.*, 796 F.2d 443, 449-450, 230 USPQ 416, 420 (Fed. Cir. 1986)).

As such, we believe no statements or support for what was the ordinary skill level in the art have been placed in the examination record by the Examiner. We believe failure to establish this *Graham inquiry* criterion is an error in examination, whereas it cannot be established what would have been obvious *to do* without first knowing the *ordinary skill level* in the art of the theoretical person *who* would find these things obvious to do. We believe the question of *what* would have been obvious cannot be separated from the question of *who* would have been expected to do them.

We believe that the Examiner is a fact finder required to resolve *Graham* inquiries ("Examination Guidelines for Determining Obviousness Under 35 U.S.C. 103 in View of the Supreme Court Decision in *KSR International Co. v. Teleflex Inc.*," Fed. Reg., Vol. 72, No. 195, October 10, 2007). According to the Court in *Graham v. John Deere Co.*, 383 U.S. 1, 148 USPQ 459 (1966), it is critical to determining obviousness under 35 U.S.C. §103 to ascertain the level of ordinary skill in the art, whereas this is pivotal in the language and standard set forth in the law at 35 USC §103.

The Court reasoned in *In re Gentile* (Civ. App. No. 93-1086 (Fed Cir. Oct. 5, 1993)), that unless an explicit level of ordinary skill in the art is established during examination of a patent

application, the ordinary level can be presumed from that which is indicated by the cited art. However, we respectfully submit that the cited art is drawn from inventors of extraordinary skill in the art, and thus their teachings do not indicate what was ordinary skill at the time of Applicant's invention.

According to the USPTO's patent databases, Stern *et al.* (Messrs. Stern, Johnston and Ms. Moller) are named as inventors on 9 issued US patents to date (6,807,668; 6,535,930; 6,212,577; 6,133,918; 5,956,030; 5,835,919; 5,812,862; 5,754,178; and 5,669,005). And, according to a Google search, it appears that Mr. Mark Apperley of Waikato University has had 30 scholarly published papers in the field of computing science (see link at http://portal.acm.org/author_page.cfm?id=81100385323&coll=GUIDE&dl=portal.ACM&trk=0&CFID=27217964&CFTOKEN=33725169). Similarly, Apperley's co-author Mr. Dale Fletcher, has two published scholarly works, and Mr. Bill Rogers has 7 published scholarly works. (See links at: http://portal.acm.org/author_page.cfm?id=81100337703&coll=GUIDE&dl=portal.ACM&trk=0&CFID=27217964&CFTOKEN=33725169 and at http://portal.acm.org/author_page.cfm?id=81100173937&coll=GUIDE&dl=portal.ACM&trk=0&CFID=27217964&CFTOKEN=33725169)

We believe that using the works of authors and inventors who have among them so many issued patents and scholarly publications as an indicator of ordinary skill would be an error in examination. Whereas the Examiner has not entered any evidence according to the Graham inquiries regarding skill level, we believe an error in examination has occurred. We therefore request reversal of all rejections under 35 U.S.C. §103(a), or in the alternative, we request remand to the Examiner with instruction to perform such analysis.

Rejections of 1, 6, and 11

With respect to the rejections of Claims 1, 6, and 11, we initially responded to the examiner when these rejections were first made by respectfully disagreeing with the Examiner's reasoning, and by submitting that Apperley fails to teach *"concatenating two or more information elements into said buffer"*.

By "concatenate", we mean a process in which information which is newly "copied" is appended to the contents or information already stored in the clipboard, so that the clipboard only contains one information item at a time, but that information item represents the "collected up" information from multiple copy operations. This is consistent with the term as used in the specification (see paragraphs 0042, 0069, 0074, 0078 - 0079, and 0082), and it is consistent with the extrinsic definition of the term:

Dictionary:**concatenate -verb**

1. to link together; unite in a series or chain.

(Source: Dictionary.com Unabridged (v 1.1). Retrieved October 16, 2007,
from Dictionary.com website: [http://dictionary.reference.com
/browse/concatenate](http://dictionary.reference.com/browse/concatenate))

Referring to Apperley, section 2, paragraph 4, we submitted that we believe Apperley teaches that GNU Emacs editor keeps a *history list* of items which have been copied, but does not actually "concatenate" those items for the user (e.g. the user must make multiple selections from the list and then operate the paste command to achieve concatenation) (our emphasis added):

Apperley:

A copy and paste system that comes close to meeting our needs is that provided by the GNU Emacs text editor [8].

The paste buffer holds a **history of copied text selections**.

The 'select and paste' menu option **displays a sub-menu**

of the most recent entries. Any can be pasted into the

document. This makes it possible to copy a series of

fragments and then perform a series of pastes in some

different document or area of the same document, without the need for focus flipping. Because of the limited space available when displaying the sub-menu however, only a tiny amount of text can be displayed, and it can be difficult to recognize fragments. Further, Emacs stores all deleted text in the paste buffer. There is no distinction made between fragments deliberately cut or copied, and text that is simply being removed. As a result the paste list tends to be quite long, and somewhat unpredictable. Text removed while correcting spelling errors will be entered alongside deliberately copied segments.

And, referring to Apperley, section 3.1, paragraph 3, we submitted that we believe Apperley is silent regarding concatenating information in the clipboard, but instead is referring to enabling and disabling their "multiple paste" feature:

Apperley:

In the trial implementation the SST can be switched on and off using the tick box at the lower right of the dialog box (Figure 2). When the SST is active, the pipe and prompt serve to remind the user that they should be trying to find a particular piece of data. In effect, activating the SST constitutes the issue of a multiple paste command. It is appropriate to issue the paste before finding the data, because that is the natural order in which the user addresses the problem.

We believe that Apperley's multiple paste feature does not concatenate, but instead copies each item to a destination immediately after it is selected. For example, refer to Apperley's multiple paste buffers (e.g. more than one buffer, not concatenating in a single buffer) at section 2, paragraph 4. Apperley's disclosure does not contain any other instances of the term "multiple paste".

Referring to Apperley, section 3.3, paragraph 7, we also submitted that Apperley is silent regarding concatenating information from multiple copy commands into the single clipboard buffer, but instead is discussing implementation strategies, and even discouraging the

assumption that all copy commands are to be treated as commands to transfer information (our emphasis added):

Apperley:

There were many options for implementing the 'copy and paste' command. The application we have experimented with most often, Internet Explorer, has a COM interface through which selected data might be extracted. In fact, we have chosen to simulate a 'copy' menu selection on the grounds that the event used is common to a large number of applications. The SST software then takes the clipped text from the system clipboard and puts it into the dialog box field. There are also a number of alternative ways in which we might have implemented the 'fire' command. One possibility would have been to simply allow the user to clip using the normal command of the source application and monitor the system clipboard for incoming data. **We rejected this option for two reasons.** One was that applications are not uniform in providing short cut keys or mouse gestures for cutting and we didn't want the user to have to use a menu, with its attendant large mouse movements. **The other, and more important reason was that we couldn't assume that all copy events were intended to deliver data to our system. The user might, for example, want to copy and paste a URL as part of their searching process.**

We believe, and proposed to the Examiner, that such an assumption made by Apperley would lean away from concatenation, which by its nature requires that multiple copy commands be assumed to be a "copy and append" operation to the contents of the clipboard

We believe, in view of Apperley's dissuasion from operational assumption that all copy commands were to be handled as information transfer (and concatenation) commands, that it would have been unreasonable for one skilled in the art to set aside this portion of Apperley's disclosure, and to proceed to make the combination, substitutions, and modifications as proposed in the rationale for the rejections. For these reasons, in addition to the reasons that an ordinary

skill level had not been established, we requested allowance of claims 1, 6, and 11.

However, in the final rejections, the Examiner failed to establish the ordinary skill level, and held our arguments to be unpersuasive. The examiner stated:

In response to applicant's argument, Apperley discloses having a paste buffer that holds a history of copied information elements (text selections), wherein the actual copied information elements are appended in sequence to form a single/elongated information item of previously copied information elements (sect.2, par.4). Apperley further discloses a similar method that allows a monitoring of incoming data to a single system clipboard (sect.3.3, last par.).

We respectfully maintain our arguments, and we point out that the Examiner's entire phrase " . . . *the actual copied information elements are appended in sequence to form a single/elongated information item of previously copied information elements (sect.2, par.4).* Apperley further discloses a similar method that allows a monitoring of incoming data to a single system clipboard (sect.3.3, last par.) " is not a quote from the reference, but instead is the Examiner's interpretation of the phrase. We believe there is no support in the reference for this interpretation, and there is no affidavit averring to the Examiner's own knowledge under 37 C.F.R. §1.104(d)(2).

We respectfully point out that the Apperly reference itself complains of the "unpredictable" nature of the list referred to by the Examiner (Apperly's pg. 3, righthand column, last sentence of first full paragraph). And, we respectfully point out that Apperly states that there is no distinction made in the list between deleted and copied text (same paragraph).

We believe that an error in examination has occurred by misinterpreting the reference, and using improper hindsight to read our disclosure and claims into the reference.

"Given the subtle but powerful attraction of a hindsight-based obviousness analysis, we require a rigorous application of the requirement for a showing of the teaching or motivation to combine prior art references." *In re Beasley*, Civ. App. 04-1225, slip op. at 6-7, 2004 WL 2793170 (Fed. Cir. Dec. 7, 2004) (unpublished)

We also submitted that Apperley that we believe teaches away from the proposed combination through these negative statements in the Apperley paper. The Examiner has responded in the final rejections by pointing reasoning that Apperley discouraged methods similar to "copy and append" because of "concerns" regarding copied data compatibility, but that Stern had solved this concern. We respectfully disagree that one reference can be used to overcome the teaching away of a second reference. We do not believe case law supports this approach, and the Examiner has not supported this approach with any citations. We believe that relevant case law establishes that only one of the references must teach away from the proposed combination or modification to render the combination or modification non-obvious.

We are also unable to find anywhere in Apperley's disclosure where this particular "concern" is disclosed, and the Examiner has not provided a citation to indicate where a recitation of this reason for teaching away is found. We believe that an error in examination has occurred.

We respectfully request allowance of Claims 1, 6, and 11, as they recite steps, elements, and limitations not taught by Apperley or Stern, and whereas a *prima facie* case of obviousness has not been established for (a) failure to establish what was the ordinary skill level in the art at the time of our invention, and (b) because there can be no obviousness where one or more references teach away from the proposed combination or modifications.

Rejections of Claims 5, 10 and 15.

In the final Office Action, Claims 5, 10, and 15 were rejected over Stern in view of Apperley. Claims 5, 10, and 15 depend from Claims 1, 6, and 11, respectively. We respectfully request allowance of Claims 5, 10, and 15 because

- (a) a *prima facie* case of obviousness has not been established for failure to establish what was the ordinary skill level in the art at the time of our invention;
- (b) a *prima facie* case of obviousness has not been established because there can be no obviousness where one or more references teach away from the proposed combination or modifications; and
- (c) Stern in view of Apperley fails to teach concatenation of information in a clipboard
as described in the foregoing paragraphs.

Rejections under 35 U.S.C. §103(a) Stern in view of Apperley in Further view of Tomm in still further view of Tsuji**Rejections of Claims 2 - 4, 7-9 and 12-14**

In the final Office Action, these claims were rejected under 35 U.S.C. 103(a) as being unpatentable over Stern and Apperley in further view of Tomm et al. (Tomm, US 6,560,608 B1), and in further view of Tsuji et al. (Tsuji, US 5,586,025). Tomm was relied upon for its teaching related to selection of rules to process the copied information, and Tsuji was relied upon for its teaching related to rule management user interfaces.

However, it was not established whether Tomm or Tsuji teach concatenation of selected items in a clipboard buffer. In our first reply to these rejections, we indicated that we had word searched these disclosures and could find no instances of the term "concatenate", so we requested clarification from the Examiner. For these reasons, lacking any support in the Examiner's reasoning for where "concatenation" is taught in these additional references, we believe Stern in view of Apperley in further view of Tomm in still further view of Tsuji fails to

teach or suggest all of our claim elements, steps, and limitations.

And, we also submitted that applying so many references to so few claim elements and steps might amount to improper examination of the claims in a piecemeal fashion. The Federal Circuit has indicated that the claims must be considered as a whole, beyond analysis of only the differences between the individual claim components and multiple references:

[Although *Graham v. John Deere Co.*, 383 U.S. at 17,148 USPQ at 476, requires that certain factual inquiries, among them the differences between the prior art and the claimed invention, be conducted to support a determination of the issue of obviousness, the actual determination of the issue requires an evaluation in the light of the findings in those inquiries of the obviousness of the claimed invention as whole, not merely the differences between the claimed invention and the prior art. *Lear Siegler, Inc. v. Aeroquip Corp.*, 733 F.2d 881,221 USPQ 1025,1033 (Fed. Cir. 1984) (emphasis added). See also *Fromson v. Advance Offset Plate, Inc.*, 755 F.2d 1549,225 USPQ 26, 31 (Fed. Cir. 1985)

And:

It is impermissible to use the claimed invention, as an instruction manual or "template" to piece together the teachings of the prior art so that the claimed invention is rendered obvious. This court has previously stated that "[o]ne cannot use hindsight reconstruction to pick and choose among isolated disclosures in the prior art to deprecate the claimed invention." *In re Fritch*, 972 F.2d 2160, 23 USPQ2d 1780, 1784 (Fed. Cir. 1992) (quoting *In re Fine*, 837 F.2d 1071, 1075, 5 USPQ2d 1596, 16 (Fed. Cir. 1988)). See also *Akzo N.V. v. United States Int'l Trade Comm'n*, 808 F.2d 1471, 1 USPQ2d 1241, 1246 (Fed. Cir. 1986), *cert. denied*, 483 U.S. 909 (1987).

With respect to ordinary skill level, we also believe this proposed combination cannot be relied upon as an indication of what was ordinary skill in the art at the time of our invention. As discussed in the foregoing paragraphs, Apperly *et al.* and Stern *et al.* appear to have extraordinary skill level, not just ordinary skill level, based on a large number of issued patents and published scholarly works.

We further submit that Tomm *et al.* (Messrs. Tomm and Leckie) are not of ordinary skill level, whereas the USPTO issued patent database indicates they are named as inventors on at least 4 patents (6,757,739; 6,662,237; 6,560,608; and 6,246,403). And, we believe that Tsuji *et al.* (Tsuji, Taniguchi, Hirose, Aoyama, Tomohiro and Wakayama) are potentially named as inventors or co-inventors on as many as 106 issued patents. For this latter search, we used the USPTO database search term:

in/Tsuji-Hiroshi OR in/Taniguchi-Yoji OR
in/Hirose-Tadashi OR in/Aoyama-Toyozo OR
in/Tomohiro-Shuzo OR in/Wakayama-Satoshi

We respectfully submit that errors in examination have occurred in the rejections of these claim:

- (a) a *prima facie* case of obviousness has not been established for failure to establish what was the ordinary skill level in the art at the time of our invention and for potentially relying upon the works of inventors and authors of likely over 100 patents and scholarly works as indicators of the ordinary skill level without performing an explicit *Graham* inquiry;
- (b) a *prima facie* case of obviousness has not been established because there can be no obviousness where one or more references teach away from the proposed combination or modifications, because Apperley teaches away from the proposed combination with Stern;
- (c) Stern in view of Apperley in further view of Tomm and Tsuji fails to teach concatenation of information in a clipboard; and
- (d) the application of so many references to our claims may indicate an improper use of hindsight to examine our claims in a piecemeal fashion, considering only the differences between the cited art, and without considering our claims as a whole.

For these reasons, we respectfully request allowance of all pending claims.

Respectfully,



Robert H. Frantz
U.S. Patent Agent, Reg. N^o 42,553
Tel: (405) 812-5613
Franklin Gray Patents, LLC

Franklin Gray Patents, LLC
P.O. Box 23324
Oklahoma City, OK 73127
Tel: 405-812-5613
Fax: 405-440-2465

Claims Appendix
per 37 CFR §41.37(c)(1)(viii)

Clean Form of Amended Claims

1. A method for transferring content from one computer resource to another computer resource, comprising the steps of:
 - performing by a user in a first user interface to a first computer resource a designation of a destination point or area in said first computer resource;
 - subsequently, performing by said user in a second user interface to a second computer resource a selection of two or more information elements in said second computer resource;
 - subsequent to said step of selection, automatically copying said selected information elements into a transfer buffer, thereby concatenating two or more information elements into said buffer, said transfer buffer comprising a clipboard in memory;
 - upon attempt to automatically transfer said information items from said buffer, intercepting the transfer to said destination point or area of one or more information elements;
 - performing a compatibility check for each intercepted information element with the destination computer resource by consulting one or more user-configurable compatibility rules to classify elements as incompatible or compatible;
 - for each incompatible element, performing a compatibility handling action as defined by one or more conversion rules; and
 - for each compatible element, allowing transfer of the unmodified compatible element to the destination.

2. The method as set forth in Claim 1 further comprising the steps of:
 - invoking a rule management user interface responsive to finding no existing compatibility rule for an element to be transferred; and
 - allowing, via said rule management user interface, a user action selected from the list of creating a new compatibility rule, deleting a compatibility rule, and modifying a compatibility rule.
3. The method as set forth in Claim 1 further comprising the steps of:
 - invoking a rule management user interface responsive to finding no existing conversion rule for an element to be transferred; and
 - allowing, via said rule management user interface, a user action selected from the list of creating a new conversion rule, deleting a conversion rule, and modifying a conversion rule.
4. The method as set forth in Claim 3 wherein said user action of creating and modifying a conversion rule comprises creating and modifying a conversion rule which specifies performing an action selected from the group of
 - converting a text element from one format to another format, converting a graphic image element from one format to another format, converting a video clip element from one format to another format, converting an audio clip element from one format to another format, converting animated image element from one format to another format, isolating an element, isolating an element and transferring an annotation to said destination, isolating an element and transferring a hyperlinked annotation to said destination.

5. The method as set forth in Claim 1 wherein said step of performing a compatibility handling action comprises performing an action selected from the list of converting a text element from one format to another format, converting a graphic image element from one format to another format, converting a video clip element from one format to another format, converting an audio clip element from one format to another format, converting animated image element from one format to another format, isolating an element, isolating an element and transferring an annotation to said destination, isolating an element and transferring a hyperlinked annotation to said destination.
6. A computer readable medium encoded with software for transferring content from one computer resource to another computer resource, said software performing the steps of:
 - performing by a user in a first user interface to a first computer resource a designation of a destination point or area in said first computer resource;
 - subsequently, performing by said user in a second user interface to a second computer resource a selection of two or more information elements in said second computer resource;
 - subsequent to said step of selection, automatically copying said selected information elements into a transfer buffer, thereby concatenating two or more information elements into said buffer, said transfer buffer comprising a clipboard in memory;
 - upon attempt to automatically transfer said information items from said buffer, intercepting the transfer to said destination point or area of one or more information elements selected;
 - performing a compatibility check for each intercepted information element with the destination computer resource by consulting one or more user-configurable compatibility rules to classify elements as incompatible or compatible;
 - for each incompatible element, performing a compatibility handling action as defined by one or more conversion rules; and
 - for each compatible element, allowing transfer of the unmodified compatible element to the destination.

7. The computer readable medium as set forth in Claim 6 further comprising software for performing the steps of:
 - invoking a rule management user interface responsive to finding no existing compatibility rule for an element to be transferred; and
 - allowing, via said rule management user interface, a user action selected from the list of creating a new compatibility rule, deleting a compatibility rule, and modifying a compatibility rule.
8. The computer readable medium as set forth in Claim 6 further comprising software for performing the steps of:
 - invoking a rule management user interface responsive to finding no existing conversion rule for an element to be transferred; and
 - allowing, via said rule management user interface, a user action selected from the list of creating a new conversion rule, deleting a conversion rule, and modifying a conversion rule.
9. The computer readable medium as set forth in Claim 8 wherein said software for creating and modifying a conversion rule comprises software for creating and modifying a conversion rule which specifies performing an action selected from the group of converting a text element from one format to another format, converting a graphic image element from one format to another format, converting a video clip element from one format to another format, converting an audio clip element from one format to another format, converting animated image element from one format to another format, isolating an element, isolating an element and transferring an annotation to said destination, isolating an element and transferring a hyperlinked annotation to said destination.

10. The computer readable medium as set forth in Claim 6 wherein said software for performing a compatibility handling action comprises software for performing an action selected from the list of converting a text element from one format to another format, converting a graphic image element from one format to another format, converting a video clip element from one format to another format, converting an audio clip element from one format to another format, converting animated image element from one format to another format, isolating an element, isolating an element and transferring an annotation to said destination, isolating an element and transferring a hyperlinked annotation to said destination.
11. A system for automatically transferring content from one computer resource to another computer resource, said system having one or more circuits, one or more programs executed by a processor, or a combination of circuits and processor-executed programs comprising:
 - a transfer buffer comprising a clipboard in memory;
 - a destination-first, source-second element copier configured to allow a user in a first user interface to a first computer resource to designate a destination point or area in said first computer resource, to subsequently select in a second user interface to a second computer resource two or more information elements in said second computer resource, and to subsequently automatically copy said selected information elements into said transfer buffer thereby concatenating said information elements into said buffer;
 - a transfer interceptor configured to, upon attempt to copy said information elements from said buffer, intercept one or more information elements;
 - a compatibility checker configured to verify the compatibility of each intercepted information element with the destination by consulting one or more user-configurable compatibility rules to classify elements as incompatible or compatible; and
 - a compatibility action handler configured to perform an action as defined by one or more conversion rules for each intercepted information element, and further configured to allowing transfer of said compatible elements to the destination without modification.

12. The system as set forth in Claim 11 further comprising:
 - a rule management user interface, invoked in response to finding no existing compatibility rule for an element to be transferred; and
 - one or more user options provided via said rule management user interface, for selecting a compatibility rule management action from the list of creating a new compatibility rule, deleting a compatibility rule, and modifying a compatibility rule.
13. The system as set forth in Claim 11 further comprising:
 - a rule management user interface, invoked in response to finding no existing conversion rule for an element to be transferred; and
 - one or more user options provided via said rule management user interface, for selecting a user action from the list of creating a new conversion rule, deleting a conversion rule, and modifying a conversion rule.
14. The system as set forth in Claim 13 wherein said user action list comprises at least one action selected from the list of creating and modifying a conversion rule which specifies performing an action selected from the group of converting a text element from one format to another format, converting a graphic image element from one format to another format, converting a video clip element from one format to another format, converting an audio clip element from one format to another format, converting animated image element from one format to another format, isolating an element, isolating an element and transferring an annotation to said destination, isolating an element and transferring a hyperlinked annotation to said destination.

15. The system as set forth in Claim 11 wherein said compatibility action handler is further configured to perform an action selected from the list of converting a text element from one format to another format, converting a graphic image element from one format to another format, converting a video clip element from one format to another format, converting an audio clip element from one format to another format, converting an animated image element from one format to another format, isolating an element, isolating an element and transferring an annotation to said destination, isolating an element and transferring a hyperlinked annotation to said destination.

Evidence Appendix***per 37 CFR §41.37(c)(1)(ix)***

No evidence has been submitted by applicant or examiner pursuant to 37 CFR §§1.130, 1.131, or 1.132.

Related Proceedings Appendix*per 37 CFR §41.37(c)(1)(x)*

No decisions have been rendered by a court or the Board in the related proceedings as identified under 37 CFR §41.37(c)(1)(ii).